REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
07-04-2015	Final Report	5-Sep-2014 - 4-Mar-2015

4. TITLE AND SUBTITLE

Final Report: Parallel Two-Electron Reduced Density Matrix Based Electronic Structure Software for Highly Correlated Molecules and Materials

5a. CONTRACT NUMBER

5b. GRANT NUMBER

W911NF-14-P-0032

5c. PROGRAM ELEMENT NUMBER

665502

6. AUTHORS

Eugene DePrince

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES AND ADDRESSES

Q-Chem, Inc.
6601 Owens Drive
Suite 105
Pleasanton, CA 94588 -0000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS (ES)

U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSOR/MONITOR'S ACRONYM(S)
ARO

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

65925-CH-ST1.1

12. DISTRIBUTION AVAILIBILITY STATEMENT

Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

The views, opinions and/or findings contained in this report are those of the author(s) and should not contrued as an official Department of the Army position, policy or decision, unless so designated by other documentation.

14. ABSTRACT

Variational two-electron reduced-density-matrix (2-RDM) methods provide a reference independent description of the electronic structure of many-electron systems; these methods also naturally capture multireference correlation effects. However, existing implementations of the variational 2-RDM (v-2RDM) method exhibit a time-to-solution that is often prohibitively large for the general description of large molecules or materials. It is the goal of this STTR to develop an efficient, parallel, and commercializable 2-RDM-based electronic structure software suite. In Phase I of this STTR, we have: (i) developed a prototype commercializable software product to perform electronic

15. SUBJECT TERMS

parallel variational 2-RDM semidefinite solver

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	15. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			A. Eugene DePrince
UU	UU	UU	UU		19b. TELEPHONE NUMBER: 850-645-1291

Report Title

Final Report: Parallel Two-Electron Reduced Density Matrix Based Electronic Structure Software for Highly Correlated Molecules and Materials

ABSTRACT

Variational two-electron reduced-density-matrix (2-RDM) methods provide a reference independent description of the electronic structure of many-electron systems; these methods also naturally capture multireference correlation effects. However, existing implementations of the variational 2-RDM (v-2RDM) method exhibit a time-to-solution that is often prohibitively large for the general description of large molecules or materials. It is the goal of this STTR to develop an efficient, parallel, and commercializable 2-RDM-based electronic structure software suite. In Phase I of this STTR, we have: (i) developed a prototype commercializable software product to perform electronic structure computations using the variational 2-RDM method, (ii) developed two semidefinite programming solvers and identified which is more likely to be of use for solving the variational 2-RDM problem for general systems, (iii) interfaced this solver with the libraries of the Q-Chem electronic structure package, (iv) implemented algorithms that use either loop-based or libtensor-based tensor manipulation and identified the libtensor algorithm as the more promising for massively parallel variational 2-RDM computations, (v) developed a pilot distributed-memory parallel version of the solver through an interface between libtensor and the Cyclops Tensor Framework (CTF), (vi) enabled the automatic detection and use of graphics processing units (GPUs) for certain portions of the algorithm, and (vii) extended some of our algorithms for the description of open-shell systems and to include partial three-particle N-representability conditions.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

Received Paper

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations

A. Eugene DePrince III, “Electronic excitation energies from a time-dependent two-electron reduced-density-matrix method.” 55th Sanibel Symposium, St. Simon’s Island, GA (February 15, 2015).

Number of Presentations: 1.00

Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Non Peer-Reviewed Conference Proceeding publications (other than abstracts):

Peer-Reviewed Conference Proceeding publications (other than abstracts):

Received Paper

TOTAL:

Number of Peer-Reviewed Conference Proceeding publications (other than abstracts):

(d) Manuscripts

Received Paper

TOTAL:

Number of Manuscripts:

Books

Received Book

TOTAL:

Received Book Chapter

TOTAL:

Patents Submitted

Patents Awarded

Awards

Graduate Students

NAME	PERCENT_SUPPORTED	Discipline
Daniel Nascimento	0.67	
FTE Equivalent:	**0.67**	
Total Number:	**1**	

Names of Post Doctorates

NAME	PERCENT_SUPPORTED
Jacob Fosso-Tande	1.00
FTE Equivalent:	**1.00**
Total Number:	**1**

Names of Faculty Supported

NAME	PERCENT_SUPPORTED	National Academy Member
A. Eugene DePrince	0.00	
FTE Equivalent:	**0.00**	
Total Number:	**1**	

Names of Under Graduate students supported

NAME	PERCENT_SUPPORTED
FTE Equivalent:	
Total Number:	

Student Metrics

This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00

The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields:..... 0.00

Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale):..... 0.00

Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering:..... 0.00

The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense 0.00

The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields: 0.00

Names of Personnel receiving masters degrees

NAME
Total Number:

Names of personnel receiving PHDs

NAME
Total Number:

Names of other research staff

NAME	PERCENT_SUPPORTED
Yihan Shao	0.20
Evgeny Epifanovsky	0.45
Zhengting Gan	0.20
FTE Equivalent:	**0.85**
Total Number:	**3**

Sub Contractors (DD882)

Inventions (DD882)

Scientific Progress

See attachment.

Technology Transfer

STTR Phase I Final Report: Parallel Two-Electron Reduced Density Matrix Based Electronic Structure Software for Highly Correlated Molecules and Materials

I. LIST OF APPENDICES, ILLUSTRATIONS AND TABLES

II. STATEMENT OF THE PROBLEM STUDIED

Quantum mechanical methods based upon the N-electron wave function are plagued by unfavorable computational scaling with respect to system size, and an exact solution to the Schrödinger equation (within a given basis set) formally requires computational effort that scales *exponentially* with system size. For this reason, the full configuration interaction (CI) method is only applicable to systems having roughly a dozen electrons. Many methods do scale more favorably with system size (e.g. polynomially), but oftentimes these methods have been developed for the accurate description of only dynamical electron correlation. The treatment of nondynamical (or static) electron correlation is challenging, and often requires a complete-active-space (CAS) self-consistent-field (SCF) description of some reference space. Unfortunately, CAS computations amount to performing a full CI computation within the active space, the cost of which increases exponentially with the size of the active space. Methods that employ the two-electron reduced-density matrix (2-RDM) as the central variable (instead of the wave function) have the potential to overcome this scaling problem.

Variational 2-RDM (v-2RDM) methods[1–5] provide a reference independent description of the electronic structure of many-electron systems; these methods also naturally capture static correlation effects. However, existing implementations of the v-2RDM method exhibit a time-to-solution that is often prohibitively large for the general description of large molecules or materials.

It is the goal of this STTR to develop an efficient, parallel, and commercializable 2-RDM-based electronic structure software suite. In Phase I of this STTR, we have: (i) developed a prototype commercializable software product to perform electronic structure computations using the variational 2-RDM method, (ii) developed two semidefinite programming solvers and identified which is more likely to be of use for solving the variational 2-RDM problem for general systems, (iii) interfaced this solver with the libraries of the Q-Chem[6] electronic structure package, (iv) implemented algorithms that use either loop-based or libtensor[7]-based tensor manipulation and identified the libtensor algorithm as the more promising for massively parallel variational 2-RDM computations, (v) developed a pilot distributed-memory parallel version of the solver through an interface between libtensor and the Cyclops Tensor Framework (CTF),[8] (vi) enabled the automatic detection and use of graphics processing units (GPUs) for certain portions of the algorithm, and (vii) extended some of our algorithms for the description of open-shell systems and to include partial three-particle N-representability conditions.[5]

III. SUMMARY OF THE MOST IMPORTANT RESULTS

III A. Phase I objectives

Our stated objectives of Phase I were threefold:

- **implementing and benchmarking a boundary-point semidefinite programming (SDP) solver[9–11] for the variational 2-RDM method.** We would compare the performance of this algorithm to the matrix-factorization-based algorithm we used to generate our preliminary data in the Phase I proposal.

- **interfacing our v-2RDM code with libraries from the Q-Chem electronic structure package[6] to develop a new commercializable 2-RDM-based software suite**. A particular emphasis would be placed on implementing the v-2RDM method using Q-Chem's advanced tensor library, libtensor.[7]

- **developing a graphics processing unit (GPU)-enabled and a prototype distributed-memory parallel variational 2-RDM algorithm.** The initial parallel implementation would be acheived through an interface between libtensor and the Cyclops Tensor Framework (CTF).[8]

We describe below how each of these objectives was met during the Phase I award period. We also describe additional milestones achieved that were not explicitly stated as part of the Phase I work.

III B. Phase I accomplishments

Objective 1: Developing a boundary point semidefinite programming solver

Entering Phase 1, we had already developed a matrix-factorization-based semidefinite programming solver for the v-2RDM method based on the implementation described in Ref. 12. The code was implemented as a plugin to the Psi4 electronic structure package.[13] The algorithm minimized the electronic energy with respect to the elements of the 2-RDM subject to the set of two-particle

N-representability[14] conditions known as the DQG (or PQG) conditions.[15] The first goal of our Phase I effort was to implement a different solver, called a boundary-point semidefinite solver, and to benchmark the performance of this algorithm relative to the matrix-factorization-based algorithm. In the first month of Phase I, we implemented the boundary-point solver (again enforcing the DQG conditions) as a plugin to Psi4. Table I illustrates the relative performance of the boundary-point and matrix-factorization algorithms for several small closed-shell molecules represented by the STO-3G basis set. The boundary-point algorithm is clearly the superior algorithm;

TABLE I: Total time (s) required for variational 2-RDM optimizations using two different semidefinite programming algorithms.

	time (s)		
molecule	matrix factorization	boundary point	speedup
Be	15	1	15
LiH	2506	2	1253
BH	12246	9	1361
H_2O	992	4	248
N_2	31445	28	1123
HF	267	2	133
CO	670	3	223
HCN	4715	7	674

for several molecules, the time to solution is more than 1000 smaller than that required by the matrix-factorization-based algorithm. Hence, we have focused all of our subsequent development efforts on the more promising boundary-point algorithm. We should note here that there are minor technical differences in how the 2-RDM is represented within the boundary-point and matrix-factorization-based algorithms. In generating the data in Table I, our boundary-point algorithm used spin-adapted basis functions[16] for the various representations of the 2-RDM, and the matrix-factorization-based algorithm did not. Spin adaptation does boost the efficiency of the algorithm, but the majority of the speedups presented in Table I can be attributed to the efficiency of the boundary-point semidefinite programming model relative to that of the matrix-factorization-based model.

Objective 2: Interfacing our variational 2-RDM code with Q-Chem libraries.

We next interfaced our boundary-point semidefinite solver with the low-level libraries of the Q-Chem electronic structure package. The initial phase of this work was completed in the second month of the Phase I award. The v-2RDM method can now be invoked using either a Q-Chem-style input file [Fig. 1(a)] or using the IQmol molecular visualizer [Fig. 1(b)]. To set up a v-2RDM computation within IQmol, a user can construct a molecule using the graphical interface and simply change the default method from HF ("Hartree-Fock") to RDM. Currently, only the default options for the v-2RDM method are enabled through the IQmol interface; in Phase II, we will modify IQmol so as to allow a user to select various 2-RDM-related options through pull-down menus. The options available through the text interface control various convergence criteria and the positivity conditions to be enforced ("DONLY," "DQ," "DQG,", etc.). Once a v-2RDM computation has completed, some of the output generated can be interpreted and visualized by IQmol. For example, Figure 2 illustrates one of the natural orbitals (the second-highest occupied natural orbital, or HONO-1) for a water molecule computed using the v-2RDM/STO-3G method.

(a)
```
$molecule
     0 1                    ! a neutral singlet
     O
     H 1 1.0
     H 1 1.0 2 104.5
$end

$rem
     correlation     RDM    ! selects the variational 2-RDM method
     basis           6-31G  ! the one-electron basis set
     rdm_positivity  DQG    ! enforces DQG two-positivity conditions
     rdm_eps_convergence 4  ! convergence in the constraint errors
     rdm_e_convergence   4  ! convergence in the primal/dual energy gap
     rdm_cg_convergence  6  ! convergence in the conjugate gradient procedure
$end
```

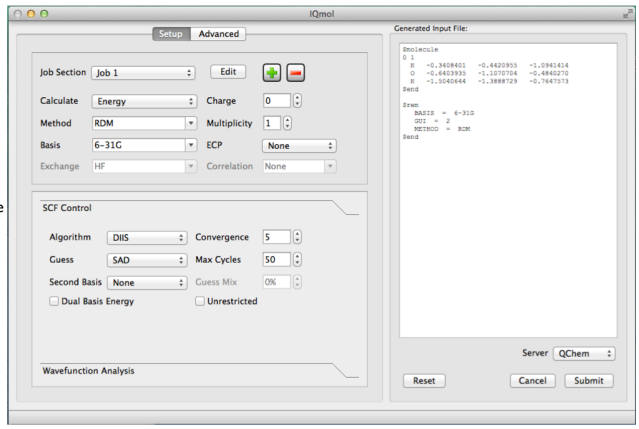
(b)


FIG. 1: Computations using the v-2RDM method can be specified using either (a) a Q-Chem-style text input file or (b) the IQmol molecular viewer.

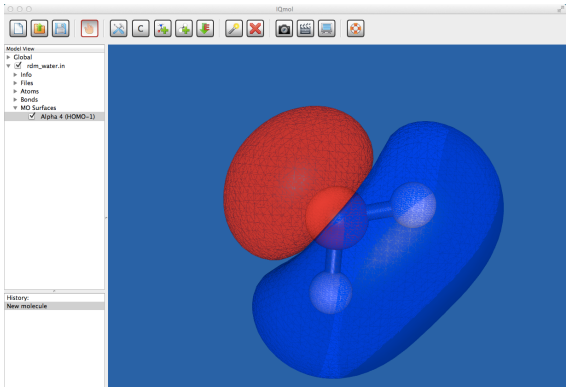


FIG. 2: The IQmol molecular viewer can display natural orbitals from the variational 2-RDM method.

The initial implementation of the boundary-point v-2RDM method interfaced with the Q-Chem librares handled the N-representability constraints and all related tensor manipulations explicitly in low-level C++ code, therefore providing direct access to tuning the program for performance. Our second implementation codes the N-representability constraints using the high-level tensor expression language of the libtensor library. This library was designed for rapid development of tensor-algebra-based computational algorithms. Performance tuning of this second implementation must be done at the level of libtensor's computational kernels. In order to fully appreciate the relative performance of the two implementations, we must first review the general structure of the boundary-point semidefinite solver (for more details on the semidefinite algorithm, please see Refs. 9–11). Three kernels dominate the cost of the solver: (1) the action of the constraint matrix, $\mathbf{A}$, on the primal solution vector $\mathbf{x}$ [$\mathcal{O}(k^4)$], (2) the action of the transpose of the constraint matrix, $\mathbf{A}^{\mathrm{T}}$, on the dual solution vector, $\mathbf{y}$ [$\mathcal{O}(k^4)$], and (3) a diagonalization and transformation step to update the primal solution [$\mathcal{O}(k^6)$]. Here, k represents the dimension of the one-electron basis set. Kernel 3 is evaluated on the order of 1000 times for a given optimization, and kernels 1 and 2 are typically evaluated 50,000 to 100,000 times per optimization. Hence, for small systems, kernels 1 and 2 tend to dominate the cost of the computation, despite their fourth-power scaling.

Figure 3 shows a timing comparison of the $\mathbf{A}\cdot\mathbf{x}+\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ and diagonalization/transformation steps of the algorithm for increasing system size (the x-axis represents the log of the total number of basis functions). The $\mathbf{A}\cdot\mathbf{x}$ and $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ kernels were evaluated 1000 times and the diagonalization/transformation kernel was evaluated 10 times. This ratio of kernel calls reflects that of a typical v-2RDM optimization. In this log-log plot the slope of each line corresponds to the power of the computational cost scaling. The hand-coded algorithm exhibits correct scaling (k^4) for all problem sizes. The libtensor-based implementation is slower at the onset, but becomes faster than

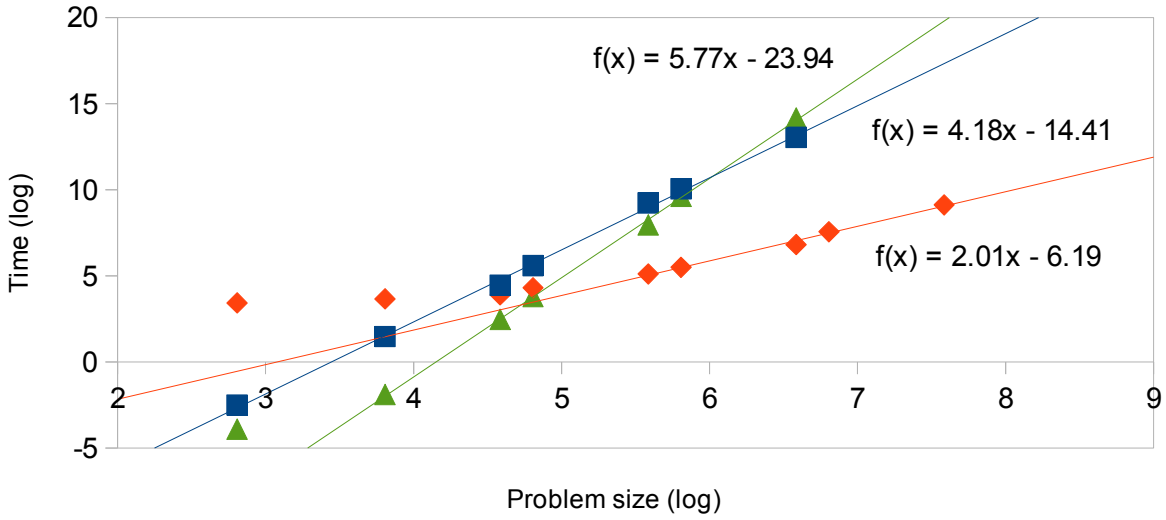


FIG. 3: Timing comparison of the $\mathbf{A}\cdot\mathbf{x}+\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ and diagonalization/transformation steps of the BPSDP algorithm. Two implementations of $\mathbf{A}\cdot\mathbf{x}+\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ are the hand-written code (blue) and using libtensor (red). The diagonalization/transformation step is shown in green.

the hand-written code at about 30 orbitals, and significantly faster for larger problems. Its apparent scaling of k^2 indicates that, for problem sizes in the considered range, the computation is dominated by steps that have a lower scaling (and a very large prefactor) than the algorithm as a whole; this invites the opportunity for further performance enhancements. The diagonalization/transformation step scales close to the sixth power in problem size, as expected, and starts to dominate the calculation with 30 orbitals and larger.

Objective 3: Node-level and distributed-memory parallelism

We now consider the shared-memory parallel scaling of the $\mathbf{A}\cdot\mathbf{x}$, $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$, and diagonalization/transformation steps. Timings for 1000 evaluations of the $\mathbf{A}\cdot\mathbf{x}$ and $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ steps and 10 calls to the diagonalization/transformation step are given in seconds, with scaling factors in parentheses. These computations were performed on an Intel Xeon E5-2690 system (2 processors with 8 cores each). To study the strong scaling properties of the algorithm, we considered a system with 192 orbitals. Timings and scaling factors are provided in Table II. At present, the parallel speedup for the $\mathbf{A}\cdot\mathbf{x}$ and $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ steps does not exceed 2–2.5 even at full CPU load. The theoretical peak performance for memory-bound algorithms on this computer system would yield a 4–6x speedup.

The weak scaling properties of the steps of the algorithm are shown in Fig. 4. For each increase in problem size, a proportional increase in the number of computing processors was used. With perfect parallel scaling, the polynomial degree of the cost of each step would be reduced by one. We observe perfect scaling for the diagonalization/transformation step, a CPU bound problem. The $\mathbf{A}\cdot\mathbf{x}$ and $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ steps scale poorly as they are memory-bound problems. However, for all problem sizes considered, except for the smallest, the diagonalization/transformation step dominates the entire computation and therefore is the best candidate for parallelization.

TABLE II: Parallel scaling of the $\mathbf{A}\cdot\mathbf{x}$, $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ and diagonalization/transformation steps. Timings for 1000 iterations of $\mathbf{A}\cdot\mathbf{x}$ and $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ steps and 10 iterations of diagonalization/transformation are given in seconds, scaling factors in parentheses. These computations were performed on an Intel Xeon E5-2690 system (2 processors with 8 cores each).

processors	$\mathbf{A}\cdot\mathbf{x}$	$\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$
1	366	191
2	223 (1.6)	143 (1.3)
4	173 (2.1)	127 (1.5)
8	144 (2.5)	115 (1.7)
16	138 (2.7)	115 (1.7)

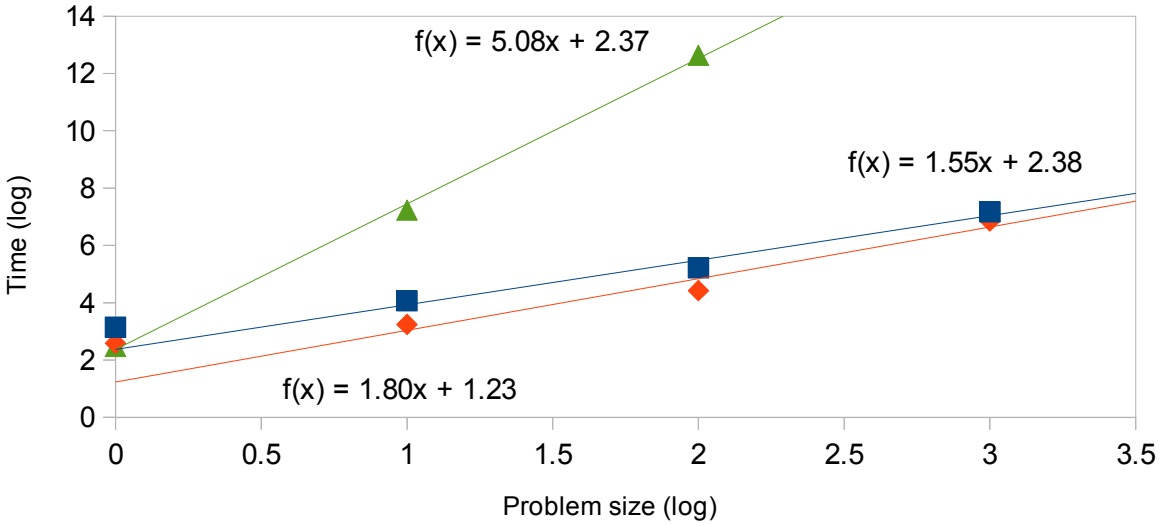


FIG. 4: Weak scaling properties of the $\mathbf{A}\cdot\mathbf{x}$ (blue), $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ (red) and diagonalization/transformation (green) steps in the libtensor implementation of the v-2RDM method.

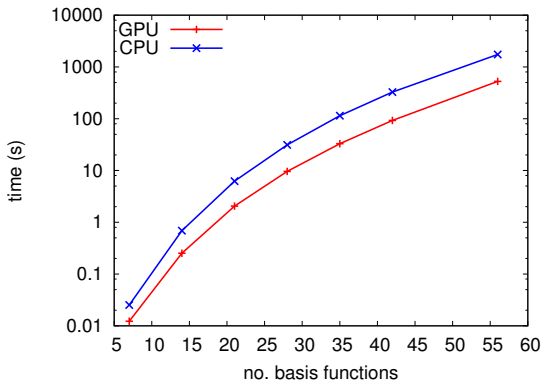


FIG. 5: GPU and CPU timings (s) for the diagonalization/transformation step of the boundary-point algorithm.

Our pilot distributed-memory implementation is based on libtensor's use of the Cyclops Tensor Framework (CTF),[8] a distributed tensor algebra library, as a back-end. While we succeeded in developing a functioning distributed-memory parallel algorithm using CTF, our preliminary results show that the current version of the code does not achieve strong or weak scaling in the $\mathbf{A}\cdot\mathbf{x}$ and $\mathbf{A}^{\mathrm{T}}\cdot\mathbf{y}$ steps. The cause of this failure is to be investigated further during Phase II, and possible reasons include inefficiencies in interfacing libtensor with CTF and inefficiencies within CTF, neither of which were specifically optimized for this use case. Nonetheless, in the regime of large problems, which is the target of the distributed memory implementation, the diagonaliza-

tion/transformation step of the algorithm strongly dominates the total walltime. This step will be more easily parallelizable, and during Phase II we will use one of the very efficient existing distributed parallel eigensolvers (e.g. ScaLAPACK, PLAPACK, Elemental) for this step.

We have also explored the possibility of accelerating the diagonalization/transformation kernel through the use of graphics processing units (GPUs), which are automatically detected and utilized by our software (if the software is configured with a "cuda" option). Figure 5 provides the time required to evaluate the diagonalization/transformation step 10 times for systems of varying sizes. This step is evaluated using either a single core of an Intel Core i7 3930k CPU or an NVIDIA Tesla K40c (Kepler) GPU. The GPU is utilized through the culaDsyev call of the CULA linear algebra library (for the diagonalization) and the cublasDgemm call of the NVIDIA CUDA Basic Linear Algebra Subroutine (cuBLAS) library (for the transformation). For the systems considered, the GPU is as much as 3.5 times more efficient than the CPU. This comparison is not completely fair in that the CPU code is utilizing only one of six physical cores. However, it does demonstrate the ability of the software to detect and utilize GPUs to accelerate this portion of the algorithm. Further, these computations consider two-particle N-representability conditions, and dimensions of the matrices being diagonalized are not large enough to fully exploit the GPU. As is discussed below, we have also implemented a set of partial three-particle N-representability conditions[5] for which this step would involve the diagonalization and transformation of much larger matrices. We will investigate this case in Phase II, and we expect the utility of GPUs to become more apparent.

IV. ADDITIONAL MILESTONES

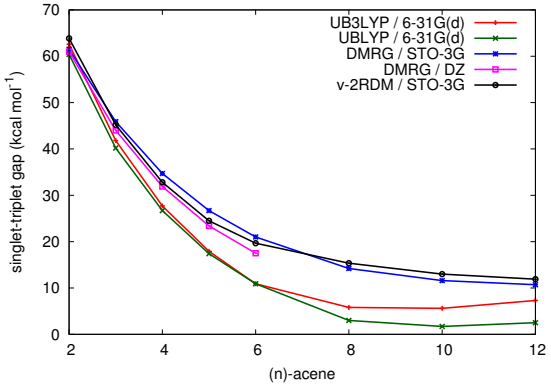


FIG. 6: Singlet/triplet gaps (kcal mol^{-1}) for acene molecules of increasing length. DMRG, UBLYP, and UB3LYP results results are taken from Ref. 17.

To this point, we have only considered spin-adapted, closed-shell implementations of the two-particle N-representability conditions known as the DQG (or PQG) conditions. We have also generalized our hand-written (non-libtensor) code to perform open-shell optimizations. Unlike the closed-shell implementation, the open-shell implementation does not use spin-adapted basis functions. Upon inspection of the spin block structure of each of the representations of the 2-RDM (particularly the ${}^2G$ matrix) for non-singlet states, it becomes clear that spin adaptation is not does not yield much computational savings for open-shells,[18] and a much simpler algorithm can be designed without the use of spin-adapted basis functions. One must then explicitly constrain the expectation value of $\hat{S}^2$ in order to maintain a pure spin state,[16] but this constraint is quite easily enforced.

We recently applied our open-shell variational 2-RDM code to the evaluation of singlet/triplet gaps in the linear acene molecules.[19] We have explored the performance of the v-2RDM method with the DQG two-particle N-representability conditions for molecules as large as to 12-acene (dodecacene) represented by the STO-3G basis set. Our computations can be thought of as a complete-active-space (CAS) calculation where only the orbitals of the π-network are considered as active (the occupations of all other occupied and virtual orbitals are held fixed at their Hartree-Fock values). The largest computation (12-acene) is comparable to a complete active space computation

with 50 electrons in 50 orbitals. This system size is clearly beyond what can be treated using conventional configuration-interaction-based CAS methods. Figure 6 illustrates the singlet/triplet gap computed at the variational 2-RDM (DQG), density matrix renormalization group (DMRG), and density functional theory (DFT) levels of theory. The DFT computations use the B3LYP and BLYP functionals and a 6-31g(d) basis set. DMRG and DFT results are taken from Ref. 17.

We see that the 2-positivity conditions (DQG) yield singlet/triplet gaps in these systems that are qualitatively similar to those obtained from DMRG computations. Both DMRG and variational 2RDM computations suggest that the singlet/triplet gap decreases monotonically with increasing system size and should converge to some finite value. This behavior differs from the DFT methods that suggest the singlet/triplet gap actually increases between 10- and 12-acene. **To our knowledge, these data represent the largest direct comparison of the quality of variational 2-RDM methods to other methods for a chemically meaningful open-shell systems.** We also present in Fig. 7 the natural orbital occupations for the singlet state of each molecule in the linear acene series. We observe the same emergence of polyradical behavior reported in Ref. 17 at the DMRG level of theory, but we note that the v-2RDM method tends to over correlate the electrons. This overcorrelation manifests itself in a greater degree of occupation degeneracy between the highest occupied and lowest unoccupied natural orbitals.

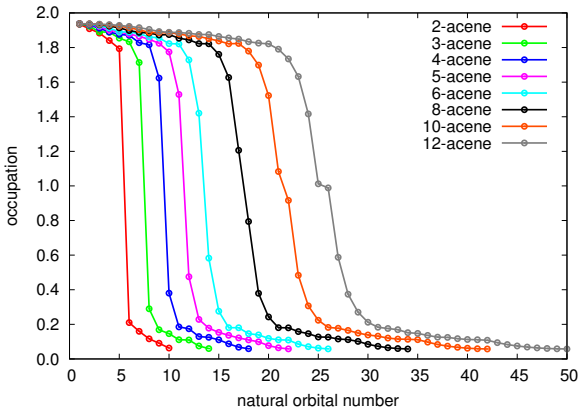


FIG. 7: Natural orbital occupations for the ground state of linear polyacenes obtained from the variational 2-RDM (DQG) level of theory.

While singlet/triplet gaps from the v-2RDM method do display excellent qualitative agreement with those obtained from DRMG, the absolute deviations from the DMRG values can be as large as 2 kcal mol^{-1}. Fortunately, the v-2RDM method is systematically improvable through the enforcement of additional N-representability conditions. Accordingly, we recently implemented stronger partial three-particle N-representability conditions known as the T1 and T2 conditions.[5] Figure 8 illustrates the error (in milihartrees, mH) for various N-representability conditions relative to full CI computations for the N_2 dissociation using the STO-6G basis set. Clearly the T2 condition greatly improves the quality of the potential energy curve. The error near the equilibrium bond length is roughly 1 mH, and the largest errors over the curve are only about 5 mH. We have only implemented these conditions in our plugin to Psi4, but interfacing the code with Q-Chem libraries should be straightforward at this point. Our Phase II proposal will include efforts to implement these T1 and T2 conditions in an efficient way using libtensor. Note that these conditions are currently expressed using non-spin-adapted basis functions and are generalized to treat both open- and closed-shell systems.

V. CONCLUSIONS

Over the past six months, we have achieved every milestone outlined in our Phase I proposal. We successfully implemented a boundary-point semidefinite solver in Month 1 that was demonstrated to be 15-1360 times more efficient than our initial matrix-factorization-based algorithm. It is clear that the boundary-point solver is the solver of choice for moving forward with this STTR. In Month

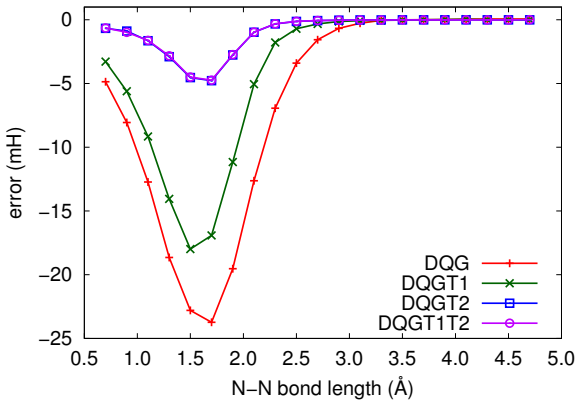


FIG. 8: Errors in the electronic energy (mH) obtained from variational 2-RDM computations with various positivity conditions throughout the N_2 dissociation curve. The basis set is STO-6G, and the errors are determined relative to full configuration interaction computations.

2, we interfaced our boundary-point solver with the low-level libraries of the Q-Chem electronic structure package. In Month 3, we reimplemented the $\mathbf{A} \cdot \mathbf{x}$ kernel of the semidefinite solver using the libtensor library and demonstrated that the libtensor solver could be substantially more efficient than the hand-written solver for large system sizes. In Month 4, as we continued reimplementing the boundary-point solver, we implemented hand-optimized open-shell semidefinite solvers. As was demonstrated above, the variational 2-RDM method with the DQG conditions can be used to obtain a qualitatively correct description of the decrease in the singlet/triplet gap in linear acene molecules of increasing size. In month 5, we finished reimplementing a pilot version of the semidefinite solver in Q-Chem using the libtensor library. Again, for large systems, the evaluation of the $\mathbf{A} \cdot \mathbf{x}$ and $\mathbf{A}^{\mathrm{T}} \cdot \mathbf{y}$ kernels is more efficient when using the libtensor library. We spent Month 6 optimizing the libtensor code and enabling point-group symmetry in that algorithm (we should note that all of our algorithms exploit abelian point-group symmetry). We have presented preliminary scaling data for both shared-memory parallelism and have developed a pilot distributed-memory parallel code using a combination of libtensor and CTF. We have also demonstrated that our software can utilize graphics processing units to accelerate at least one portion of the boundary-point semidefinite programming solver. We have also implemented two partial three-positivity conditions that substantially increase the quality of the v-2RDM method. Part of Phase II would involve reimplementing these conditions using libtensor.

VI. BIBLIOGRAPHY

[1] C. Garrod, M. V. Mihailovic, and M. Rosina, J. Math. Phys. **16**, 868 (1975); M. V. Mihailovic and M. Rosina, Nucl. Phys. A **237**, 221 (1975); M. Rosina and C. Garrod, J. Comput. Phys. **18**, 300 (1975); R. M. Erdahl, C. Garrod, B. Golli, and M. Rosina, J. Math. Phys. **20**, 1366 (1979); R. M. Erdahl, Rep. Math. Phys. **15**, 147 (1979).

[2] D. A. Mazziotti and R. M. Erdahl, Phys. Rev. A **63**, 042113 (2001).

[3] M. Nakata, H. Nakatsuji, M. Ehara, M. Fukuda, K. Nakata, and K. Fujisawa, J. Chem. Phys. **114**, 8282 (2001).

[4] D. A. Mazziotti, Phys. Rev. A **65**, 062511 (2002).

[5] Z. Zhao, B. J. Braams, M. Fukuda, M. L. Overton, and J. K. Percus, J. Chem. Phys. **120**, 2095 (2004).

[6] Y. Shao, L. Fusti-Molnar, Y. Jung, J. Kussmann, C. Ochsenfeld, S. T. Brown, A. T. B. Gilbert, L. V. Slipchenko,S. V. Levchenko, D. P. O'Neill, R. A. Distasio Jr., R. C. Lochan, T. Wang, G. J. O.

Beran, N. A. Besley, J. M., Herbert, C. Y. Lin, T. Van Voorhis, S. H. Chien, A. Sodt, R. P. Steele, V. A. Rassolov, P. E. Maslen, P. P. Korambath, R. D. Adamson, B. Austin, J. Baker, E. F. C. Byrd, H. Dachsel, R. J. Doerksen, A. Dreuw, B. D. Dunietz, A. D. Dutoi, T. R. Furlani, S. R. Gwaltney, A. Heyden, S. Hirata, C.-P. Hsu, G. Kedziora, R. Z. Khalliulin, P. Klunzinger, A. M. Lee, M. S. Lee, W. Liang, I. Lotan, N. Nair, B. Peters, E. I. Proynov, P. A. Pieniazek, Y. M. Rhee, J. Ritchie, E. Rosta, C. D. Sherrill, A. C. Simmonett, J. E. Subotnik, H. L. Woodcock III, W. Zhang, A. T. Bell, A. K. Chakraborty, D. M. Chipman, F. J. Keil, A. Warshel, W. J. Hehre, H. F. Schaefer III, J. Kong, A. I. Krylov, P. M. W. Gill, M. Head-Gordon, Phys. Chem. Chem. Phys. **8**, 3172 - 3191 (2006).

[7] E. Epifanovsky, M. Wormit, T. Ku, A. Landau, D. Zuev, K. Khistyaev, P. Manohar, I. Kaliman, A. Dreuw, and A. I. Krylov, J. Comput. Chem. **34**, 2293 (2013).

[8] E. Solomonik, D. Matthews, J. Hammond, and J. Demmel, *Cyclops Tensor Framework: reducing communication and eliminating load imbalance in massively parallel contractions*, Tech. Rep. UCB/EECS-2012-210 (EECS Department, University of California, Berkeley, 2012).

[9] J. Povh, F. Rendl, A. Wiegele, Computing **78**, 277 (2006).

[10] J. Malick, J. Povh, F. Rendl, and A. Wiegele, SIAM J. Optim. **20**, 336 (2009).

[11] D. A. Mazziotti, Phys. Rev. Lett. **106**, 083001 (2011).

[12] D. A. Mazziotti, ESAIM: Mathematical Modelling and Numerical Analysis **41**, 249 (2007).

[13] Turney, J. M.; Simmonett, A. C.; Parrish, R. M.; Hohenstein, E. G.; Evangelista, F. A.; Fermann, J. T.; Mintz, B. J.; Burns, L. A.; Wilke, J. J.; Abrams, M. L.; Russ, N. J.; Leininger, M. L.; Janssen, C. L.; Seidl, E. T.; Allen, W. D.; Schaefer, H. F.; King, R. A.; Valeev, E. F.; Sherrill, C. D.; Crawford, T. D. "Psi4: an open-source *ab initio* electronic structure program", WIREs Comput. Mol. Sci. **2**, 556-565 (2012).

[14] A. J. Coleman, Rev. Mod. Phys. 35, **668** (1963).

[15] C. Garrod and J. K. Percus, J. Math. Phys. **5**, 1756 (1964).

[16] G. Gidofalvi and D. A. Mazziotti, Phys. Rev. A **72**, 052505 (2005).

[17] J. Hachmann, J. J. Dorando, M. Avils, and G. K.-L. Chan, J. Chem. Phys. **127**, 134309 (2007).

[18] H. van Aggelen, B. Verstichel, P. Bultinck, D. Van Neck, and P. W. Ayers, The Journal of Chemical Physics **136**, 014110 (2012).

[19] J. Fosso-Tande, D. R. Nascimento, and A. E. DePrince III, in preparation (2015).